UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/647,106	08/25/2003	Andrew James Booker	550-462	9839

23117 7590 07/17/2009

NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
CHEN, QING	
ART UNIT	PAPER NUMBER
2191	
MAIL DATE	DELIVERY MODE
07/17/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/647,106	BOOKER ET AL.	
	Examiner	Art Unit	
	Qing Chen	2191	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 May 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-4,6-17,19-29 and 31-34* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4,6-17,19-29 and 31-34* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is in response to the amendment filed on May 11, 2009, entered by the RCE filed on the same date.

2. **Claims 1-4, 6-17, 19-29, and 31-34** are pending.

3. **Claims 1-4, 6-11, 13-17, 19-24, 26-29, and 31-34** have been amended.

4. **Claims 5, 18, 30, and 35-42** have been canceled.

5. The objection to the title is maintained in view of Applicant's amendments to the title and further explained hereinafter.

6. The objections to Claims 1-4, 6-17, and 19-27 are withdrawn in view of Applicant's amendments to the claims.

7. The 35 U.S.C. § 112, first paragraph, rejections of Claims 27-29 and 31-34 are withdrawn in view of Applicant's amendments to the claims.

8. The 35 U.S.C. § 112, second paragraph, rejections of Claims 27-29 and 31-34 are withdrawn in view of Applicant's amendments to the claims.

Continued Examination Under 37 CFR 1.114

9. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on May 11, 2009 has been entered.

Response to Amendment

Specification

10. The title of the invention is not descriptive. A new title is required that is clearly indicative of the invention to which the claims are directed.

The following title is suggested: GENERATING SOFTWARE TEST INFORMATION FOR SOFTWARE CODE WHEN TESTED ON A TARGET PROCESSOR.

Claim Objections

11. **Claims 3, 4, 7-9, 12, 16, 17, 20-22, 25, 29, and 31-34** are objected to because of the following informalities:

- **Claims 3, 4, 16, 17, and 29** recite the limitation "said original instruction." Applicant is advised to change this limitation to read "said at least one original instruction" for the purpose of providing it with proper explicit antecedent basis.
- **Claims 7, 8, 20, 21, 32, and 33** recite the limitation "said instruction groups." Applicant is advised to change this limitation to read "said number of instruction groups" for the purpose of providing it with proper explicit antecedent basis.
- **Claim 9** depends on Claim 7 and, therefore, suffers the same deficiency as Claim 7.
- **Claim 22** depends on Claim 20 and, therefore, suffers the same deficiency as Claim 20.
- **Claim 34** depends on Claim 32 and, therefore, suffers the same deficiency as Claim 32.

- **Claims 12 and 25** recite the limitation "said sequence of generated instructions." Applicant is advised to change this limitation to read "said corresponding sequence of generated instructions" for the purpose of providing it with proper explicit antecedent basis.
- **Claim 31** recites the limitation "said sequence of instructions." Applicant is advised to change this limitation to read "said sequence of original instructions" for the purpose of providing it with proper explicit antecedent basis.
- **Claim 31** recites the limitation "a sequence of generated instructions." Applicant is advised to change this limitation to read "a corresponding sequence of generated instructions" for the purpose of keeping the claim language consistent throughout the claims.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

12. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

13. **Claims 14-17, 19-29, and 31-34** are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 14 recites the limitation "said predetermined generated instructions." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "said predetermined generated instruction" for the purpose of further examination.

Claims 15-17 and 19-26 depend on Claim 14 and, therefore, suffer the same deficiency as Claim 14.

Claim 27 recites the limitation "a computer readable storage medium containing computer readable instructions." The claim language fails to clearly point out the details relating to how computer readable instructions are "contained" on a computer readable storage medium. Such an ambiguity further renders the claim scope indefinite for at least the reason that computer readable instructions can only be stored, recorded, or encoded on a computer readable storage medium. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "a computer readable storage medium storing computer readable instructions" for the purpose of further examination. Applicant is respectfully requested for further clarification of the claim language used.

Claims 28, 29, and 31-34 depend on Claim 27 and, therefore, suffer the same deficiency as Claim 27.

Claim 29 recites the limitation "said status information." There is insufficient antecedent basis for this limitation in the claim. In the interest of compact prosecution, the Examiner subsequently interprets this limitation as reading "status information" for the purpose of further examination.

Claim Rejections - 35 USC § 102

14. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

15. **Claims 27-29 and 31** are rejected under 35 U.S.C. 102(a) as being anticipated by **Applicant Admitted Prior Art (hereinafter "AAPA")**.

As per **Claim 27**, AAPA discloses:

- a) generating, from a sequence of original instructions, at least one original instruction of said sequence of original instructions including a condition code, a corresponding sequence of generated instructions as said software test instructions, wherein for selected original instructions having said condition code, a corresponding generated instruction is a predetermined generated instruction having a generated opcode including said condition code, wherein said predetermined generated instruction is an instruction which is not recognised by said target processor *(see Figure 3; Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the*

computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14."; Page 2: 3 and 4, "The instruction may be conditional and, in which case, may contain a condition code."; Page 6: 14-25, "The processor core 20 retrieves the first instruction of the generated opcode 16. The processor core 20 determines whether that instruction is a special instruction and, if so, then the handler routine 30 is invoked; otherwise, the processor core 20 executes the instruction and then retrieves the next instruction.").

As per **Claim 28**, the rejection of **Claim 27** is incorporated; and AAPA further discloses:

- wherein each instruction of said sequence of original instructions includes a condition code *(see Page 2: 3 and 4, "The instruction may be conditional and, in which case, may contain a condition code.").*

As per **Claim 29**, the rejection of **Claim 27** is incorporated; and AAPA further discloses:

- wherein said condition code is an instruction qualifier which prevents said at least one original instruction from being executed by a target processor unless status information satisfies said condition code *(see Page 2: 5-8, "The condition code indicates the conditions that those flags that must satisfy for the associated instruction to be executed. Such condition codes include EQ/NE (equal/not equal), CS/CC (carry set/carry clear), PL/MI (positive/negative), AL (always), etc.").*

As per **Claim 31**, the rejection of **Claim 27** is incorporated; and AAPA further discloses:

- generating, from said sequence of original instructions, a corresponding sequence of generated instructions, a predetermined generated instruction being generated for each instruction in the sequence of original instructions *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.").*

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

17. **Claims 1-4, 6, 10-17, 19, and 23-26** are rejected under 35 U.S.C. 103(a) as being unpatentable over **AAPA** in view of **US 4,246,637 (hereinafter "Brown")**.

As per **Claim 1**, AAPA discloses:

- a) generating, from a sequence of original instructions, at least one original instruction of said sequence of original instructions including a condition code, a corresponding sequence of generated instructions, wherein for selected instructions having said condition code, a corresponding generated instruction is a predetermined generated instruction having a generated opcode including said condition code *(see Figure 3; Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14."; Page 2: 3 and 4, "The instruction may be conditional and, in which case, may contain a condition code.")*;

- b) executing, on a target processor, said corresponding sequence of generated instructions and thereby producing software test information *(see Page 6: 10-13, "The original opcode 14 together with the generated opcode 16 is stored in the memory 22. A handler routine 30 is also stored in the memory 22 which is operable by the processor core 20 to generate code coverage and profiling information using the program code 14 and the generated opcode 16.")*; and

- c) when during said step (b) said predetermined generated instruction is encountered, determining using a check performed upon said condition code within said generated opcode with reference to status information associated with an operation of said target processor whether the corresponding condition code of said predetermined generated instruction is satisfied and, if

so, replacing said predetermined generated instruction with said corresponding original instruction from said sequence of original instructions so as to cause said corresponding original instruction to be executed, wherein said predetermined generated instruction is an instruction which is not recognised by said target processor *(see Page 6: 14-25, "The processor core 20 retrieves the first instruction of the generated opcode 16. The processor core 20 determines whether that instruction is a special instruction and, if so, then the handler routine 30 is invoked; otherwise, the processor core 20 executes the instruction and then retrieves the next instruction. Once activated, the handler routine 30 refers to the original opcode 14 and checks the condition code of the corresponding original instruction. The handler routine 30 then determines whether the original instruction would have been executed by comparing its condition code with the current status flags of the processor core 20. If the handler routine 30 determines that the original instruction would have been executed, then it replaces the special instruction with the original instruction. The handler routine 30 is then exited. The processor core 20 hardware can then execute the original instruction which has just replaced the special instruction.").*

However, AAPA does not disclose:

- determining using a check performed in hardware by said target processor upon said condition code.

Brown discloses:

- determining using a check performed in hardware by a target processor upon a condition code *(see Column 6: 6-12, "During this initial selection sequence, the host processor 11 also checks the status of the controller 10 by looking at the condition code present on the*

three-bit Condition Code In bus 43. This condition code is provided by the controller condition code register 72. The contents of register 72 reflect the current status for the controller 10.").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Brown into the teaching of AAPA to modify AAPA to include determining using a check performed in hardware by said target processor upon said condition code. The modification would be obvious because one of ordinary skill in the art would be motivated to utilize a faster hardware solution to perform code checks without the need for additional software routines.

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- wherein each instruction of said sequence of original instructions includes a condition code *(see Page 2: 3 and 4, "The instruction may be conditional and, in which case, may contain a condition code.").*

As per **Claim 3**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- wherein said condition code is an instruction qualifier which prevents said at least one original instruction from being executed by said target processor unless said status information satisfies said condition code *(see Page 2: 5-8, "The condition code indicates the conditions that those flags that must satisfy for the associated instruction to be executed. Such condition codes include EQ/NE (equal/not equal), CS/CC (carry set/carry clear), PL/MI (positive/negative), AL (always), etc.").*

As per **Claim 4**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- wherein said status information is predetermined architectural state associated with said target processor and said condition code specifies a status of said predetermined architectural state that must be met in order for said at least one original instruction to be executed *(see Page 2: 8-14, "Hence, by way of example, in the ARM (trademark) instruction set, the condition code EQ/NE requires that the zero condition flag (`Z` flag) must be set/cleared respectively for the instruction to be executed; the Z flag is set if the result of the last condition flag setting instruction was zero. Similarly, the condition code PL/MI requires that the negative condition flag (`N` flag) must be cleared/set respectively for the instruction to be executed; the N flag is set if the result of the last condition flag setting instruction was negative.").*

As per **Claim 6**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- generating, from said sequence of original instructions, a corresponding sequence of generated instructions, a predetermined generated instruction being generated for each instruction in the sequence of original instructions *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.").*

As per **Claim 10**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- incrementing a coverage counter when the condition code of the predetermined generated instruction is satisfied to provide an indication that said corresponding original instruction will be executed *(see Page 7: 10-14, "If the handler routine 30 determines that the original instruction would have been executed, then it replaces the special instruction with the original instruction, increments a counter associated with that instruction and replaces the preceding instruction with a special instruction.").*

As per **Claim 11**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- incrementing a counter associated with said corresponding original instruction when the condition code of the predetermined generated instruction is satisfied to provide an indication that said corresponding original instruction will be executed *(see Page 7: 10-14, "If the handler routine 30 determines that the original instruction would have been executed, then it replaces the special instruction with the original instruction, increments a counter associated with that instruction and replaces the preceding instruction with a special instruction.").*

As per **Claim 12**, the rejection of **Claim 11** is incorporated; and AAPA further discloses:

- replacing a preceding instruction in said corresponding sequence of generated instructions with said predetermined generated instruction having a condition code corresponding to said preceding instruction *(see Page 7: 10-14, "If the handler routine 30 determines that the original instruction would have been executed, then it replaces the special*

instruction with the original instruction, increments a counter associated with that instruction and replaces the preceding instruction with a special instruction.").

As per **Claim 13**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- executing said corresponding original instruction on said target processor *(see Page 6: 24 and 25, "The processor core 20 hardware can then execute the original instruction which has just replaced the special instruction.").*

Claims 14-17, 19, and 23-26 are apparatus claims corresponding to the method claims above (Claims 1-4, 6, and 10-13) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-4, 6, and 10-13.

18. **Claims 7-9 and 20-22** are rejected under 35 U.S.C. 103(a) as being unpatentable over **AAPA** in view of **Brown** as applied to Claims 1 and 14 above, and further in view of **US 5,712,996 (hereinafter "Schepers")**.

As per **Claim 7**, the rejection of **Claim 1** is incorporated; and AAPA further discloses:

- a2) generating said predetermined generated instruction for one instruction in each of said instruction groups *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the*

instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.").

However, AAPA and Brown do not disclose:

- a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions.

Schepers discloses:

- a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions *(see Abstract, "In order to be able to execute rapid processing of a program on super-scalar microprocessors, the individual instructions of this program must be divided into instruction groups, which can be processed by processing units of the microprocessor, in such a way that the instructions can be processed in parallel.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Schepers into the teaching of AAPA to modify AAPA to include a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions. The modification would be obvious because one of ordinary skill in the art would be motivated to execute rapid processing of the program code *(see Schepers – Abstract).*

As per **Claim 8**, the rejection of **Claim 7** is incorporated; and AAPA further discloses:

- generating said predetermined generated instruction for the last instruction in each of said instruction groups *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.").*

As per **Claim 9**, the rejection of **Claim 7** is incorporated; however, AAPA and Brown do not disclose:

- wherein said predetermined generated instruction provides information relating to the number of instructions in a corresponding instruction group of said number of instruction groups.

Schepers discloses:

- wherein said predetermined generated instruction provides information relating to the number of instructions in a corresponding instruction group of said number of instruction groups *(see Column 2: 63-65, "The number of the components per instruction group is fixed in accordance with the number of the instructions which a microprocessor can load simultaneously.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Schepers into the teaching of AAPA to modify AAPA to include wherein said predetermined generated instruction provides information

relating to the number of instructions in a corresponding instruction group of said number of instruction groups. The modification would be obvious because one of ordinary skill in the art would be motivated to determine the number of instructions a microprocessor can load simultaneously *(see Schepers – Column 2: 63-65)*.

Claim 20 is rejected for the same reason set forth in the rejection of Claim 7.

Claim 21 is rejected for the same reason set forth in the rejection of Claim 8.

Claim 22 is rejected for the same reason set forth in the rejection of Claim 9.

19. **Claims 32-34** are rejected under 35 U.S.C. 103(a) as being unpatentable over **AAPA** in view of **Schepers**.

As per **Claim 32**, the rejection of **Claim 27** is incorporated; and AAPA further discloses:

- a2) generating said predetermined generated instruction for one instruction in each of said instruction groups *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.")*.

However, AAPA does not disclose:

- a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions.

Schepers discloses:

- a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions *(see Abstract, "In order to be able to execute rapid processing of a program on super-scalar microprocessors, the individual instructions of this program must be divided into instruction groups, which can be processed by processing units of the microprocessor, in such a way that the instructions can be processed in parallel.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Schepers into the teaching of AAPA to modify AAPA to include a1) partitioning said sequence of instructions into a number of instruction groups, each instruction group including one or more instructions. The modification would be obvious because one of ordinary skill in the art would be motivated to execute rapid processing of the program code *(see Schepers – Abstract).*

As per **Claim 33**, the rejection of **Claim 32** is incorporated; and AAPA further discloses:

- generating said predetermined generated instruction for the last instruction in each of said instruction groups *(see Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the*

instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14.").

As per **Claim 34**, the rejection of **Claim 32** is incorporated; however, AAPA does not disclose:

- wherein said predetermined generated instruction provides information relating to the number of instructions in a corresponding instruction group of said number of instruction groups.

Schepers discloses:

- wherein said predetermined generated instruction provides information relating to the number of instructions in a corresponding instruction group of said number of instruction groups *(see Column 2: 63-65, "The number of the components per instruction group is fixed in accordance with the number of the instructions which a microprocessor can load simultaneously.").*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Schepers into the teaching of AAPA to modify AAPA to include wherein said predetermined generated instruction provides information relating to the number of instructions in a corresponding instruction group of said number of instruction groups. The modification would be obvious because one of ordinary skill in the art would be motivated to determine the number of instructions a microprocessor can load simultaneously *(see Schepers – Column 2: 63-65).*

Response to Arguments

20. Applicant's arguments with respect to Claims 1 and 14 have been considered but are moot in view of the new ground(s) of rejection.

In the Remarks, Applicant argues:

a) In the first full paragraph on page 27 of the Final Rejection, the Examiner appears to indicate that the Examiner is construing the claim "corresponding condition code" in an exceptionally broad manner so as to encompass the "special instruction" disclosed in the AAPA. Applicants have amended the current independent claims to preclude such a broad interpretation by specifying that the "generated instruction" has "a generated opcode including said condition code." At least in view of these amendments to the independent claims, Applicants to do not believe it is reasonable, given the plain meaning of the claim words, to consider the substitute special instruction of AAPA as being the same as the "generated opcode including said condition code."

Examiner's response:

a) Examiner disagrees. With respect to the Applicant's assertion that it is not reasonable, given the plain meaning of the claim words, to consider the substitute special instruction of AAPA as being the same as the "generated opcode including said condition code," the Examiner respectfully submits that AAPA clearly discloses "generated opcode including said condition code" *(see Figure 3; Page 1: 22-28, "An instruction set emulator 12, which is a software program which models the operation of a particular predetermined processor, is loaded onto the*

computing device 10. Also loaded onto the computing device 10 is the original opcode 14 of the program code to be analysed. From this original opcode 14, an analysis module of the instruction set emulator 12 produces generated opcode 16. In order to produce the generated opcode 16, the analysis module substitutes a special instruction (SPI) for each instruction in the original opcode 14."; Page 2: 3 and 4, "The instruction may be conditional and, in which case, may contain a condition code."). Note that Figure 3 of AAPA clearly illustrates the generated opcode instructions. Further note that AAPA also discloses that the generated opcode instructions may contain a condition code.

Therefore, for at least the reason set forth above, the rejections made under 35 U.S.C. § 103(a) with respect to Claims 1 and 14 are proper and the rejection made under 35 U.S.C. § 102(a) with respect to Claim 27 is proper and therefore, maintained.

Conclusion

21. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Q. C./

Examiner, Art Unit 2191

/Wei Y Zhen/

Supervisory Patent Examiner, Art Unit 2191